IJNRD.ORG ISSN : 2456-4184
INTERNATIONAL JOURNAL OF NOVEL RESEARCH AND DEVELOPMENT (IJNRD) | IJNRD.ORG
An International Open Access, Peer-reviewed, Refereed Journal

A SIGN TO TEXT LANGUAGE TRANSLATOR USING OPENCV

[1]Sai Harshitha Aluru, [2]Dr. A. Obulesh, [3]Tejaswi Sattarshetty, [4]Keerthi Yalamaddi, [5]Zaib Unnisa Nayeem

[1]Student, [2]HOD, [3]Student, [4]Student, [5]Student

[1]Artificial Intelligence,
[1]Vidya Jyothi Institute Of Technology, Hyderabad, India

Abstract : This research paper introduces a project on developing an efficient Sign-to-Text Language Translator with a specific focus on Indian Sign Language (ISL), utilizing OpenCV, Computer Vision, and Neural Networks. The primary goal is to bridge communication gaps for deaf-mute individuals by providing real-time interpretation of ISL gestures, converting them into text for broader societal understanding. The system employs a two-layered approach, incorporating a Convolutional Neural Network (CNN) for accurate and real-time gesture prediction. The research involves an in-depth analysis of various ISL gestures, exploring machine learning algorithms and deep learning techniques to enhance gesture recognition accuracy. Custom data sets are generated using OpenCV, and the CNN model comprises multiple layers, including Re-LU activation, max pooling, dropout layers, and the Adam optimizer. Additionally, the methodology integrates innovative features such as finger spelling sentence formation and an Auto Correct mechanism, enhancing the overall functionality and usability of the system. With a commitment to addressing communication challenges faced by the deaf-mute community, this research strives to advance sign language translation, fostering inclusively through cutting-edge technologies.

IndexTerms **– CNN, ANN, Keras, OpenCV, Tensorflow**

I. INTRODUCTION

In recent years, the convergence of technology and sign language interpretation has marked significant progress, particularly in the pursuit of inclusive communication methods for individuals with hearing impairments. Sign languages, serving as primary modes of communication for the Deaf community, play a pivotal role in fostering social interaction, education, and professional integration. Across the globe, there are over 300 sign languages, each distinguished by its unique grammar and vocabulary, underscoring the critical need for effective tools and technologies to bridge communication gaps. This research paper delves into the advancements, challenges, and potential applications of Indian Sign Language (ISL) detection within the evolving technological landscape. The linguistic and cultural intricacies of Indian Sign Language present distinctive challenges compared to other sign languages worldwide. India's diverse population adds complexity, encompassing a rich tapestry of regional sign languages that, until recently, lacked a standardized approach. Recognizing the significance of a unified Indian Sign Language, authorities have taken a notable step forward with the publication of a standardized ISL dictionary. This research adopts a multidisciplinary approach, harnessing the power of Computer Vision and Neural Networks, specifically integrating the OpenCV module in Python and Convolutional Neural Networks (CNNs). This combination holds promise for developing an accurate and efficient solution to interpret Indian Sign Language, aiming to enhance communication and inclusivity within society through a simple yet effective Human-Computer Interface (HCI) for recognizing ISL gestures. The primary challenge addressed by this research is the development of an effective Sign-to-Text Language Translator using OpenCV, tailored specifically for Indian Sign Language (ISL). The key issue lies in bridging the communication gap between the deaf-mute community and the broader society. The scarcity of accessible means to translate ISL gestures into text poses a significant barrier for those without knowledge of sign language. While sign language is indispensable for the deaf-mute population, a substantial portion of the public faces challenges in understanding and interpreting these gestures.

This research aims to overcome this hurdle by leveraging technology to create a real-time ISL interpreter. The objective is to develop a tool that accurately interprets ISL gestures and converts them into text. Through the integration of computer vision and deep learning techniques with OpenCV, this research strives to develop a reliable and user-friendly system, fostering inclusivity and effective communication. The motivation for this research arises from a commitment to empower individuals with hearing and speech disabilities, with a specific focus on the unique context of Indian Sign Language. Recognizing the pivotal role effective communication plays in fostering understanding and empathy, this research seeks to dismantle communication barriers. The Sign-to-Text Language Translator, utilizing OpenCV, serves as a technological bridge to enhance inclusivity, ensuring that everyone, regardless of their hearing abilities, can have their voices heard and understood. While sign language is universally

acknowledged for conveying messages through recognized gestures, Indian Sign Language (ISL) presents distinct challenges. Despite advancements in Sign Language Detection, ISL detection has not witnessed proportional progress, necessitating effective solutions. The primary objective of this research is to develop a simple and user-friendly interface dedicated to detecting and translating Indian Sign Language gestures into text with the utmost accuracy. The emphasis is on creating a tool that not only addresses existing communication gaps but also serves as a benchmark for efficient ISL translation solutions. This research draws on a multidisciplinary approach, integrating expertise from computer vision, machine learning, and linguistics to create a robust and culturally sensitive solution. By leveraging technological advancements, this research aims to make significant contributions to the broader field of accessible communication tools for the Deaf community.

II. LITERATURE SURVEY.

A diverse array of research contributions that significantly advance the understanding and application of computer vision and machine learning techniques in gesture and sign language recognition. Each work explores unique aspects, methodologies, and datasets, providing valuable insights into the complexities and innovations within this field. In [1],Chen et al. propose a real-time hand gesture recognition system that integrates advanced finger segmentation techniques. The paper likely delves into the specifics of the segmentation method employed and discusses the system's ability to recognize hand gestures in real-time scenarios, possibly using computer vision and image processing methodologies. Aiswarya et al.'s work in [2]focuses on sign language to speech conversion using Hidden Markov Models (HMMs). This approach likely involves training HMMs to model sign language gestures and subsequently converting these gestures into speech. The paper may provide insights into the accuracy and effectiveness of the HMM-based conversion system. In [3], Yang et al. present a sign language recognition system that leverages a Weighted Hidden Markov Model. The weighted aspect suggests a nuanced approach to recognition, possibly assigning different importance levels to various features or gestures. The paper likely details the methodology behind this weighting mechanism and its impact on recognition performance. Pal and Kakade's work in [4]explores dynamic hand gesture recognition employing the Kinect sensor. The paper may discuss the advantages of using the Kinect sensor for capturing dynamic gestures in three dimensions, potentially elaborating on how the sensor contributes to improved accuracy and robustness in recognizing gestures. Verma et al.'s paper in [5] delves into the broader context of hand gesture recognition, likely discussing the methodology employed for recognizing a variety of hand gestures. The authors may detail the features used for recognition and the overall performance of the system in recognizing diverse gestures. Bantupalli and Xie, in [6], address American Sign Language recognition] utilizing deep learning and computer vision. The paper may cover the architectural details of the deep learning model employed, as well as how computer vision techniques contribute to recognizing intricate gestures in American Sign Language. Geetha and Manjusha [10] detail a vision-based recognition system for Indian Sign Language alphabets and numerals, employing B-spline approximation. Pigou et al. [11]introduce sign language recognition using Convolutional Neural Networks (CNNs), and Huang et al. [13]focus on 3D CNNs for sign language recognition.

Beginning with temporal aspects, the use of 3D CNNs [13] is crucial for capturing nuanced dynamics in sign language expressions, enhancing accuracy and robustness. Carriera's work [14] introduces an action recognition model and the Kinetics dataset, offering insights into the model's architecture, innovations in action recognition, and the dataset's characteristics for benchmarking. Deng et al.'s seminal work [15] on ImageNet details the database's structure, content, and its profound impact on advancing computer vision research. Soomro et al.'s UCF101 dataset [16] and Kuehne et al.'s HMDB database [17] contribute to human action and motion recognition, providing details on dataset creation, diversity, and applications. Zhao et al.'s work [18] on robust background subtraction in the HSV color space discusses challenges and enhancements, while Chowdhury et al.'s method [19] integrates color information into background subtraction, emphasizing advantages over traditional methods. Focusing on sign language recognition, Mehreen and Mohammad's system [20] utilizes CNNs and computer vision. Details on architecture, the role of computer vision, and overall performance contribute to advancements in sign language interpretation. Shithate et al.'s work [21] in machine learning algorithms details specific approaches, training processes, and overall system performance

III. RESEARCH METHODOLOGY

This research paper presents a hands-centric methodology for a Sign-to-Text Language Translator, where signs are interpreted without relying on external devices. This approach prioritizes user comfort, aligning with the innate mode of sign language communication. The absence of artificial devices enhances cost-effectiveness, widens accessibility, and fosters inclusively. This vision-based approach ensures flexibility, making it easily integral into various devices and conducive to widespread adoption in diverse settings.

3.1 Dataset Creation

In the pursuit of a suitable dataset for our project, we encountered a scarcity of pre-existing datasets in the form of raw images that aligned with our specific requirements. Existing datasets primarily presented data in the format of RGB values, prompting us to embark on the creation of a customized dataset to cater to our needs. The following delineates the procedural steps we undertook in crafting our dataset. To generate our dataset, we employed the OpenCV library, a pivotal tool in computer vision applications. Initially, we captured approximately 800 images for each symbol within the Indian Sign Language (ISL) for training purposes, complemented by an additional 200 images per symbol earmarked for testing purposes. The dataset creation process commenced with the capture of individual frames via the webcam embedded in our system. Subsequently, we applied a Gaussian Blur Filter to each image, a critical step aimed at extracting diverse features from the visual content. This filtering process contributed to enhancing the discriminative aspects of the dataset, ultimately fortifying the training and testing phases of our sign language recognition model.

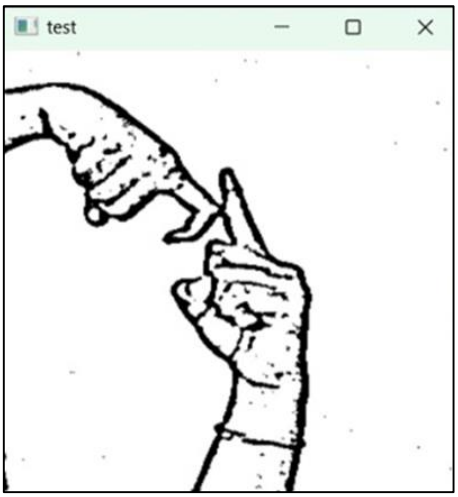

Fig.1. Gray scale image

3.2 Gesture Classification

Our research methodology employs a sophisticated dual-layered algorithmic approach with the overarching goal of predicting the user's final symbol with unparalleled precision and accuracy. In the initial layer, we leverage advanced techniques for robust symbol recognition. This encompasses the application of a Gaussian Blur filter and threshold on frames captured by OpenCV, thereby facilitating the extraction of distinctive features. Subsequently, the processed images undergo prediction through a meticulously designed Convolutional Neural Network (CNN) model. The system ensures the consistent detection of a letter over a sequence of 50 frames, triggering its printing and consideration for word formation. Furthermore, the recognition of spaces between words is seamlessly streamlined by incorporating a dedicated blank symbol into our framework.

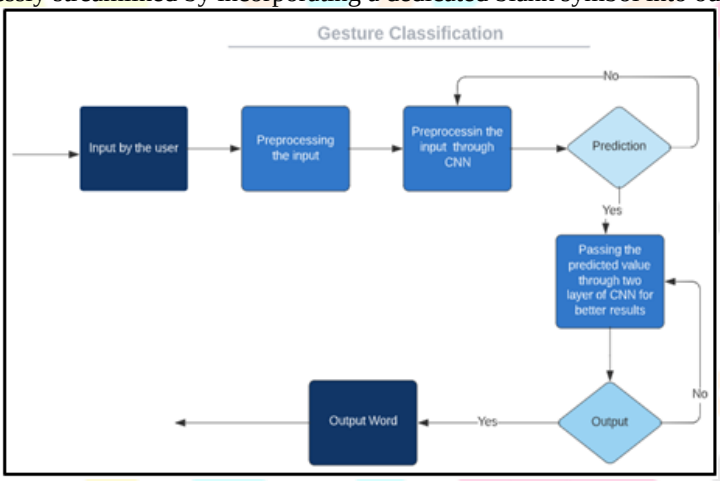

Fig.2. Sentence Prediction

Within the intricate design of Layer 1, the CNN Model plays a pivotal role in achieving efficient feature extraction and classification. The 1st Convolution Layer processes a 128x128 pixel resolution image using 32 filter weights, resulting in a 126x126 pixel image. Following this, the 1st Pooling Layer strategically downsamples the image to 63x63 pixels, followed by a 2nd Convolution Layer, further reducing it to 60x60 pixels. The 2nd Pooling Layer then refines the resolution to 30x30 pixels. The subsequent 1st Densely Connected Layer reshapes the images for input into a fully connected layer housing 128 neurons, incorporating a dropout layer (0.5) to prevent overfitting. The 2nd Densely Connected Layer transmits the output to a fully connected layer with 96 neurons, and the Final Layer is configured with the number of neurons corresponding to the classes, encompassing alphabets and the blank symbol. Each layer within this intricately designed architecture significantly contributes to the overall effectiveness of the system, ensuring precise symbol prediction and proficient feature extraction throughout the network.

To introduce nonlinearity and effectively address the vanishing gradient problem, we employ the Rectified Linear Unit (ReLU) activation function, defined as max(x, 0). This activation function not only expedites training but is complemented by max pooling with a (2, 2) pool size and ReLU activation, strategically reducing parameters and preventing overfitting. The introduction of a dropout layer is a pivotal strategy to combat overfitting, randomly deactivating activations, thereby maintaining robust classification capabilities and adaptability to diverse datasets. The Adam optimizer is strategically chosen for dynamic model updating, leveraging the benefits of ADA GRAD and RMSProp. This strategic integration optimizes the model for efficient convergence and substantially enhances real-world performance.

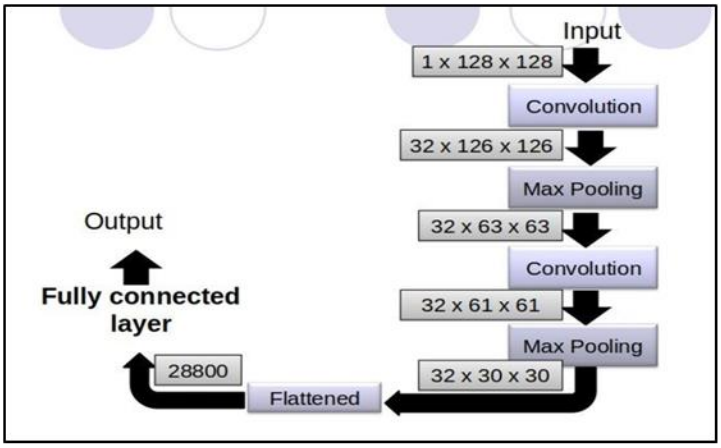

Fig.3. Model architecture

In the second layer of our algorithm, we adopt a dual-layered approach to further enhance symbol detection precision. This involves the detection of various symbol sets that may exhibit similar outcomes, followed by a meticulous classification process using specialized classifiers uniquely tailored for each set. Rigorous testing has allowed us to identify specific symbols prone to misinterpretation, leading to the development of three distinct classifiers that cater to sets with inherent similarities. This strategic and nuanced approach effectively addresses the subtleties in symbol recognition, contributing significantly to the heightened accuracy and reliability of the system's predictive capabilities.

3.3 Finger spelling sentence formation

In our detection and prediction algorithm, we implement a set of rules to ensure accurate symbol recognition and string formation. When the count of a detected letter surpasses a predetermined threshold, set at 50 in our code, and there is no other letter in close proximity within a specified threshold of 20, we initiate the printing of the detected letter and add it to the current string. This mechanism prevents premature or erroneous predictions by requiring a sustained and confident detection before updating the ongoing string. Conversely, when conditions do not meet these criteria, we clear the current dictionary storing the count of detections for the present symbol. This precautionary step minimizes the likelihood of predicting an incorrect letter, enhancing the overall accuracy of the system.

Furthermore, our algorithm incorporates a mechanism for detecting spaces. If the count of a blank (plain background) surpasses a predetermined value and the current buffer is empty, indicating no recent detections, no spaces are detected. This approach prevents unintentional space predictions in cases where the blank background count is not indicative of a deliberate space. Conversely, in scenarios where the buffer is not empty, indicating the presence of preceding symbols, the algorithm predicts the end of a word by printing a space. The current buffer is then appended to the sentence below, ensuring the accurate formation of words within the ongoing sentence. These rules collectively contribute to the robustness and reliability of our symbol detection and string formation process.

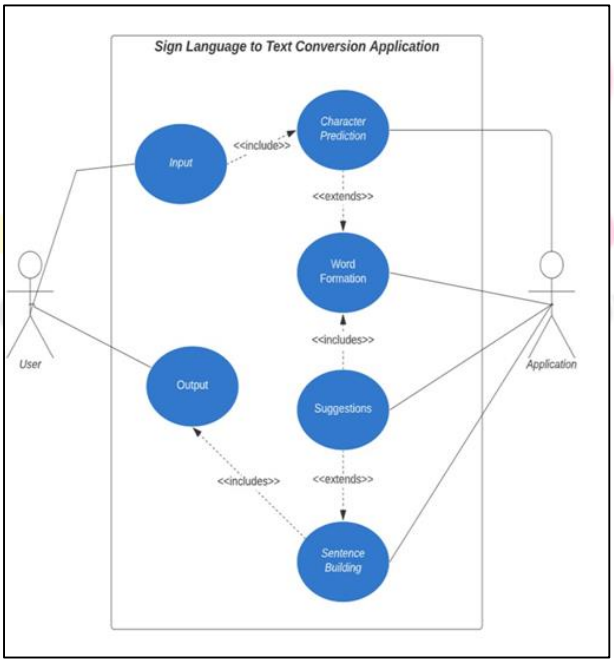

Fig.4. Sentence prediction

3.4 Auto – correction feature

In our research endeavors to refine spelling accuracy and advance the predictive capabilities of our system, we have strategically integrated the Hunspell suggest Python library. This sophisticated library plays a pivotal role by offering correct alternatives for words that may have been inaccurately input. Users benefit from a curated set of word suggestions closely matching the current word, empowering them to select the most suitable word for incorporation into the current sentence. This interactive feature not only minimizes spelling errors but also facilitates the accurate prediction of intricate words, contributing significantly to the overall effectiveness of our system. This integration serves as a valuable tool for users, providing them with the ability to choose the correct word from a set of suggestions, thereby enhancing the precision of the system and elevating the overall user experience. Through the incorporation of the Hunspell suggest library, our research aims to offer comprehensive spelling suggestions and robust support for predicting complex words within the context of the ongoing sentence.

3.5 Training and testing

The preprocessing stage is a crucial step in preparing the input data for the model. In this methodology, RGB input images are initially transformed into grayscale. This simplification not only reduces the dimensionality of the data but also retains essential features for hand gesture recognition. Following this, a Gaussian blur is applied to the images. The purpose of this blur is to mitigate unnecessary noise in the images, which can interfere with the model's ability to accurately detect and classify hand gestures. By enhancing the clarity of hand gesture features, the Gaussian blur contributes to better overall performance. An adaptive threshold is then applied to facilitate the extraction of the hand from the background. This step is essential for isolating the relevant information in the images and removing any unwanted background elements. Thresholding helps create a binary image where the hand is distinct from the background, aiding in subsequent processing steps. To ensure consistency in input size, the preprocessed images undergo resizing to a standardized resolution of 128 x 128. Standardizing the input size is crucial for the model to effectively learn and generalize patterns across different hand gestures. During both the training and testing phases, the preprocessed images serve as input for the model. The model's prediction layer is responsible for estimating the likelihood of the image belonging to predefined classes.. Normalization is accomplished through the SoftMax function, which ensures that the sum of values in each class equals 1. This normalization facilitates the interpretation of the output as probabilities, making it easier to assess the model's confidence in its predictions. Recognizing the potential deviation of the initial output from actual values, model training is conducted using labeled data to enhance predictive accuracy. The performance in classification tasks is measured using the cross-entropy function. This continuous function provides a meaningful metric by exhibiting positivity when the predicted values differ from the labeled values and zeros when they match exactly. Optimization of the model involves minimizing the cross-entropy by adjusting the weights of the neural networks. TensorFlow, the chosen framework, provides an intrinsic function for calculating cross-entropy, simplifying the implementation of this crucial step.To further enhance optimization, the Gradient Descent method is employed, specifically utilizing the Adam Optimizer. The Adam Optimizer is chosen for its efficiency in achieving convergence. It iteratively adjusts the weights of neural networks based on cross-entropy gradients, contributing to the fine-tuning of the model. This iterative adjustment process ensures that the model is trained to minimize discrepancies between predicted and actual values, ultimately improving overall accuracy and reliability in classifying hand gestures. The application of the Adam Optimizer represents a sophisticated approach to optimization, striking a balance between efficiency and effectiveness in training neural networks for the specific task at hand.

IV. RESULTS AND DISCUSSION

Our model has demonstrated remarkable efficiency, achieving a commendable accuracy of 95.8\% with the utilization of solely the first layer in our algorithm. Upon the integration of both the first and second layers, our accuracy soars to an impressive 98.0\%, surpassing benchmarks established by several contemporary research papers that specifically address Indian Sign Language (ISL). This noteworthy performance enhancement underscores the effectiveness of our comprehensive approach.A distinguishing feature of our methodology is its independence from specialized devices, such as Kinect, for hand detection. This key characteristic sets our model apart within the current landscape of research studies, providing a practical and versatile solution for hand image classification. The ability to operate without reliance on proprietary hardware not only enhances the accessibility of our approach but also contributes to its adaptability in various real-world scenarios.

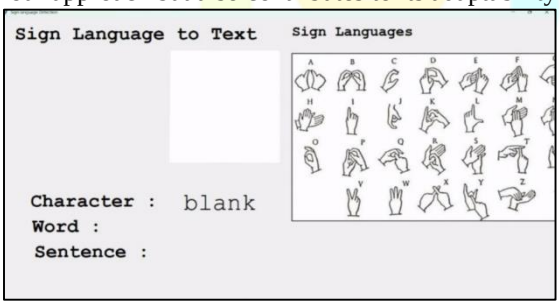

Fig.5. Detection of blank

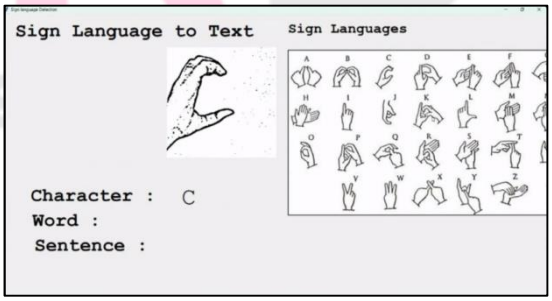

Fig.6. Detection of C

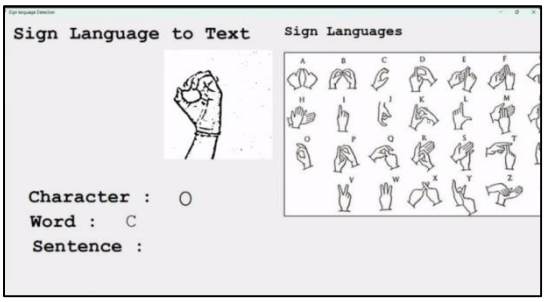

Fig.7. Detection of O

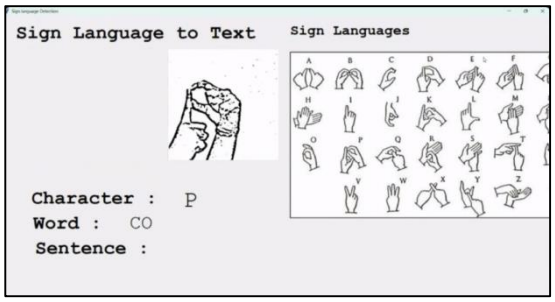

Fig.8. Detection of P

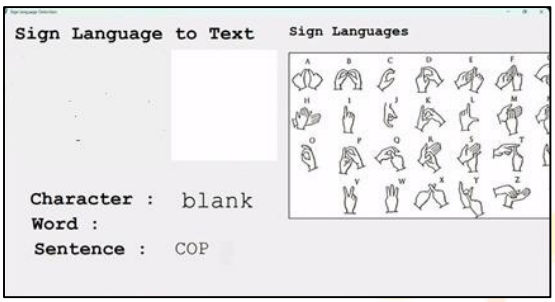

Fig.9. Detection of word

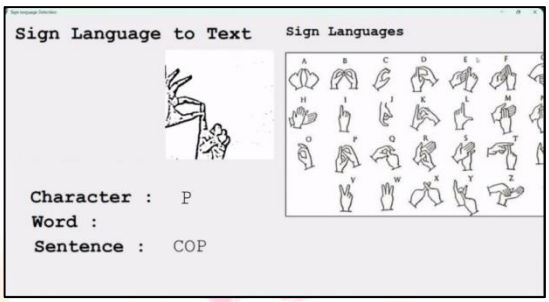

Fig.10. Detection of next word

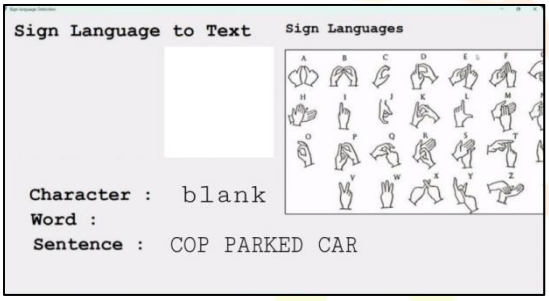

Fig.11. Complete sentence

This research introduces a robust Sign-to-Text Language Translator focused on Indian Sign Language (ISL), employing a sophisticated blend of OpenCV, Computer Vision, and Neural Networks. The two-layered approach, featuring a Convolutional Neural Network (CNN), demonstrates exceptional accuracy in real-time gesture prediction. The methodology addresses the challenges posed by the linguistic and cultural intricacies of ISL, showcasing its adaptability within India's diverse sign language landscape. The dataset creation process, utilizing OpenCV for image capture and Gaussian Blur for feature enhancement, ensures the effectiveness of the training and testing phases. The implemented gesture classification algorithm, enriched with innovative features such as finger spelling sentence formation and an Auto Correct mechanism, enhances the system's overall functionality and usability. The hands-centric methodology prioritizes user comfort, avoiding reliance on external devices and promoting cost-effectiveness and inclusivity. Notably, the model's independence from specialized devices, such as Kinect, makes it a practical and versatile solution for hand image classification in various real-world scenarios. The results highlight the model's outstanding performance, achieving an accuracy of 95.8\% with the first layer and an impressive 98.0\% with the integrated two-layer approach. This surpasses benchmarks set by contemporary research papers, reinforcing the efficacy of our comprehensive methodology. In summary, this research contributes to advancing sign language translation, addressing communication challenges faced by the deaf-mute community. The implemented technologies, coupled with innovative features, showcase the potential for widespread adoption, fostering inclusivity through cutting-edge advancements in computer vision and neural network applications.

REFERENCES

[1] Real-Time Hand Gesture Recognition Using Finger Segmentation Zhihua Chen, Jung-Tae Kim, Jianning Liang, Jing Zhang, and Yu-Bo Yuan

[2] Hidden Markov model-based Sign Language to Speech Conversion System Aiswarya V, Naren Raju N, Johanan Joy Singh S, Nagarajan T, Vijayalakshmi P

[3] Sign Language Recognition System Based on Weighted Hidden Markov Model Wenwen Yang, Jinxu Tao, Changfeng Xi, Zhongfu Yen

[4] Dynamic Hand Gesture Recognition Using Kinect Sensor Devendra kumar H. Pal, S. M. Kakade

[5] Hand Gesture Recognition Akash Verma , Gourav Soni , Pankaj Kumar Sangam , Gitanjali Ganpatrao Nikam , Mukta Rani Dhiman

[6] Y. Yorozu, M. Hirano, K. Oka, and Y. Tagawa, "Electron spectroscopy studies on magneto-optical media and plastic substrate interface," IEEE Transl. J. Magn. Japan, vol. 2, pp. 740–741, August 1987 [Digests 9th Annual Conf. Magnetics Japan, p. 301, 1982].

[7] Bantupalli, K., Xie, Y.: American sign language recognition using deep learning and computer vision. In: 2018 IEEE International Conference on Big Data (Big Data), pp 4896-4899.IEEE(2018).

[8] Cabrera, M.E., Bogado, J.M., Fermin, L., Acuna, R., Ralev, D.: Glovebased gesture recognitionsystem.In:Adaptive Mobile Robotics. Pp 747- 753 (2012).

[9] He, S.: Research of a sign language translation system based on deep learning. In: 2019 International Conference on Artificial Intelligence and Advanced Manufacturing (AIAM), pp. 392-396. IEEE (2019).

[10] Herath, H.C.M., Kumari, WAL.V., Senevirathne, W.A.PB.. Dissanayake, M.B.: Image based sign language recognition system for Sinhala sign language. Sign 3(5), 2 (2013)

[11] Geetha, M., Manjusha, U.C.: A vision based recognition of Indian sign language alphabets and numerals using b-spline approximation. Int. J. Comput. Sci. Eng 4(3), 406-415(2012)

[12] Pigou, L., Dieleman, S., Kindermans, P-J., Schrauwen, B.: Sign language recognition using convolutional neural networks. In: Agapito, L. Bronstein, M.M., Rother, C. (eds) ECCV 2014. LNCS, vol. 8925. pp. 572-578. Springer, Cham (2015).

[13] Escalera, S., et al.: ChaLearn looking at people challenge 2014: dataset and results. In Agapito, L., Bronstein, M.M., Rother, C. (eds.) ECCV 2014. LNCS, vol. 8925, pp. 459-473.Springer, Cham (2015).

[14] Huang, J., Zhou, W., Li, H.: Sign language recognition using 3D convolutional neural networks. In: IEEE International Conference on Multimedia and Expo (ICME), pp. 1-6. IEEE Turin (2015)

[15] Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., Fei-Fei, L.: ImageNet: a large-scale hierarchical image database. In: IEEE Conference on Computer Vision and Pattern Rec (CVPR 2009), pp. 248-255. IEEE. Miami, FL. USA (2009)

[16] Soomro, K., Zamir, A.R., Shah, M.: UCF101: a dataset of 101 human actions classes from videos in the wild. arXiv preprint arXiv:1212.0402(2012)

[17] Kuchne, H., Jhuang, H., Garrote, E., Poggio, T., Serre, T.: HMDB: a large video database for human motion recognition. In: 2011 IEEE International Conference on Computer Vision (ICCV), pp. 2556-2563. IEEE (2011)

[18] Zhao, M.. Bu. J., Chen, C.: Robust background subtraction in HSV color space.In: Proceedings of SPIE MSAV, vol. 1. P. 4861 (2002).

[19] Chowdhury, A., Cho, S.J., Chong, U.P: A background subtraction method using color information in the frame averaging process. In: Proceedings of 2011 6th International Forum on Strategic Technology, vol. 2, pp. 1275-1279. IEEE(2011).

[20] Mehreen, H., Mohammad, E.: Sign language recognition system using convutionalneural network and computer vision. Int. J. Eng. Res. Technol. 09(12) (2020) deeplearningbooks.org: Convolutional Networks

[21] Shithate, R.S.., Shinde,VD., Metkari, S.A., Borkar, PU. KhandGE, MA: Sign Language recognition using machine learning algorithm. Int. Res. J. Eng. Technol. 7(03),2122-2125(2020)